

## PROGRAMACIÓN DIDÁCTICA

Ciclos Formativos	Desarrollo de Aplicaciones Multiplataforma (DAM) Desarrollo de Aplicaciones WEB (DAW)	Fecha aprobación
		5-octubre-2017
Normativa que regula los títulos	DAM <ul style="list-style-type: none"> <li>Real Decreto 450/2010 de 16 de Abril</li> <li>DECRETO 3/2011, de 13 de enero, del Consejo de Gobierno, por el que se establece para la Comunidad de Madrid el currículo del ciclo formativo de grado superior correspondiente al título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma.</li> </ul>	
	DAW <ul style="list-style-type: none"> <li>Real Decreto 686/2010, de 20 de mayo</li> <li>DECRETO 1/2011, de 13 de enero, del Consejo de Gobierno, por el que se establece para la Comunidad de Madrid el currículo del ciclo formativo de grado superior correspondiente al título de Técnico Superior en Desarrollo de Aplicaciones Web.</li> </ul>	

Módulo Profesional:	Programación (código: 0485)		
Departamento	Informática y Comunicaciones	Fecha de entrada en vigor	5-octubre-2017

Duración del Módulo (en horas)	Distribución horaria semanal
270 horas	8h/semana 3-2-2-1 ó 2-2-2-2

<b>Competencia general del Ciclo Formativo DAM</b>
La competencia general de este título consiste en desarrollar, implantar, documentar y mantener aplicaciones informáticas multiplataforma, utilizando tecnologías y entornos de desarrollo específicos, garantizando el acceso a los datos de forma segura y cumpliendo los criterios de «usabilidad» y calidad exigidas en los estándares establecidos.
<b>Competencia general del Ciclo Formativo DAW</b>
La competencia general de este título consiste en desarrollar, implantar, y mantener aplicaciones web, con independencia del modelo empleado y utilizando tecnologías específicas, garantizando el acceso a los datos de forma segura y cumpliendo los criterios de accesibilidad, usabilidad y calidad exigidas en los estándares establecidos.

<b>Competencias profesionales, personales y sociales asociadas al módulo profesional.</b>
Las competencias profesionales, personales y sociales de este título son las que se relacionan a continuación: a) Configurar y explotar sistemas informáticos, adaptando la configuración lógica del sistema según las necesidades de uso y los criterios establecidos. e) Desarrollar aplicaciones multiplataforma con acceso a bases de datos utilizando lenguajes, librerías y herramientas adecuados a las especificaciones. f) Desarrollar aplicaciones implementando un sistema completo de formularios e informes que permitan

gestionar de forma integral la información almacenada.

i) Participar en el desarrollo de juegos y aplicaciones en el ámbito del entretenimiento y la educación empleando técnicas, motores y entornos de desarrollo específicos.

j) Desarrollar aplicaciones para teléfonos, PDA y otros dispositivos móviles empleando técnicas y entornos de desarrollo específicos.

t) Establecer vías eficaces de relación profesional y comunicación con sus superiores, compañeros y subordinados, respetando la autonomía y competencias de las distintas personas.

w) Mantener el espíritu de innovación y actualización en el ámbito de su trabajo para adaptarse a los cambios tecnológicos y organizativos de su entorno profesional.

**Objetivos Generales del Ciclo Formativo DAM que el módulo profesional ayuda a conseguir.**

e) Seleccionar y emplear lenguajes, herramientas y librerías, interpretando las especificaciones para desarrollar aplicaciones multiplataforma con acceso a bases de datos.

j) Seleccionar y emplear técnicas, lenguajes y entornos de desarrollo, evaluando sus posibilidades, para desarrollar aplicaciones en teléfonos, PDA y otros dispositivos móviles.

q) Seleccionar y emplear lenguajes y herramientas, atendiendo a los requerimientos, para desarrollar componentes personalizados en sistemas ERP-CRM.

w) Identificar los cambios tecnológicos, organizativos, económicos y laborales en su actividad, analizando sus implicaciones en el ámbito de trabajo, para mantener el espíritu de innovación

**Objetivos Generales del Ciclo Formativo DAW que el módulo profesional ayuda a conseguir.**

e) Interpretar el diseño lógico, verificando los parámetros establecidos para gestionar bases de datos.

j) Emplear herramientas y lenguajes específicos, siguiendo las especificaciones, para desarrollar componentes multimedia

q) Programar y realizar actividades para gestionar el mantenimiento de los recursos informáticos.

**Unidad/es de Competencia Acreditada/s por el Módulo Profesional**

UC0227\_3 Desarrollar componentes software en lenguajes de programación orientados a objetos.

## Resultados de aprendizaje/Capacidades Terminales del Módulo Profesional

### **1. Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.**

Criterios de evaluación:

- a) Se han identificado los bloques que componen la estructura de un programa
- b) informático.
- c) Se han creado proyectos de desarrollo de aplicaciones
- d) Se han utilizado entornos integrados de desarrollo.
- e) Se han identificado los distintos tipos de variables y la utilidad específica de
- f) cada uno.
- g) Se ha modificado el código de un programa para crear y utilizar variables.
- h) Se han creado y utilizado constantes y literales.
- i) Se han clasificado, reconocido y utilizado en expresiones los operadores del
- j) lenguaje.
- k) h) Se ha comprobado el funcionamiento de las conversiones de tipo explícitas
- l) e implícitas.
- m) Se han introducido comentarios en el código.

### **2. Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.**

Criterios de evaluación:

- a) Se han identificado los fundamentos de la programación orientada a
- b) objetos.
- c) Se han escrito programas simples.
- d) Se han instanciado objetos a partir de clases predefinidas.
- e) Se han utilizado métodos y propiedades de los objetos.
- f) Se han escrito llamadas a métodos estáticos.
- g) Se han utilizado parámetros en la llamada a métodos.
- h) Se han incorporado y utilizado librerías de objetos.
- i) Se han utilizado constructores.
- j) Se ha utilizado el entorno integrado de desarrollo en la creación y compilación
- k) de programas simples.

### **3. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.**

Criterios de evaluación:

- a) Se ha escrito y probado código que haga uso de estructuras de selección.
- b) Se han utilizado estructuras de repetición.
- c) Se han reconocido las posibilidades de las sentencias de salto.
- d) Se ha escrito código utilizando control de excepciones.
- e) Se han creado programas ejecutables utilizando diferentes estructuras de
- f) control.
- g) Se han probado y depurado los programas.
- h) Se ha comentado y documentado el código.

### **4. Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.**

Criterios de evaluación:

- a) Se ha reconocido la sintaxis, estructura y componentes típicos de una
- b) clase.
- c) Se han definido clases.
- d) Se han definido propiedades y métodos.
- e) Se han creado constructores.

- f) Se han desarrollado programas que instancien y utilicen objetos de las clases
- g) creadas anteriormente.
- h) Se han utilizado mecanismos para controlar la visibilidad de las clases y de
- i) sus miembros.
- j) g) Se han definido y utilizado clases heredadas.
- k) Se han creado y utilizado métodos estáticos.
- l) Se han definido y utilizado interfaces.
- m) Se han creado y utilizado conjuntos y librerías de clases.

#### **5. Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.**

Criterios de evaluación:

- a) Se ha utilizado la consola para realizar operaciones de entrada y salida de
- a) información.
- b) Se han aplicado formatos en la visualización de la información.
- c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las
- d) librerías asociadas.
- e) Se han utilizado ficheros para almacenar y recuperar información.
- f) Se han creado programas que utilicen diversos métodos de acceso al
- g) contenido de los ficheros.
- h) f) Se han utilizado las herramientas del entorno de desarrollo para crear
- i) interfaces gráficos de usuario simples.
- j) g) Se han programado controladores de eventos.
- i) Se han escrito programas que utilicen interfaces gráficos para la entrada y
- k) salida de información.

#### **6. Escribe programas que manipulen información, seleccionando y utilizando tipos avanzados de datos.**

Criterios de evaluación:

- a) Se han escrito programas que utilicen arrays.
- b) Se han reconocido las librerías de clases relacionadas con tipos de datos
- l) avanzados.
- c) Se han utilizado listas para almacenar y procesar información.
- d) Se han utilizado iteradores para recorrer los elementos de las listas.
- e) Se han reconocido las características y ventajas de cada una de la colecciones
- m) de datos disponibles.
- f) Se han creado clases y métodos genéricos.
- g) Se han utilizado expresiones regulares en la búsqueda de patrones en
- n) cadenas de texto.
- h) Se han identificado las clases relacionadas con el tratamiento de documentos
- o) XML.
- i) Se han realizado programas que realicen manipulaciones sobre documentos
- p) XML.

#### **7. Desarrolla programas, aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.**

Criterios de evaluación:

- a) Se han identificado los conceptos de herencia, superclase y subclase.
- b) Se han utilizado modificadores para bloquear y forzar la herencia de clases
- q) y métodos.
- c) Se ha reconocido la incidencia de los constructores en la herencia.
- d) Se han creado clases heredadas que sobrescriban la implementación de
- r) métodos de la superclase.
- e) Se han diseñado y aplicado jerarquías de clases.

- f) Se han probado y depurado las jerarquías de clases.
- g) Se han realizado programas que implementen y utilicen jerarquías de
- s) clases.
- h) Se ha comentado y documentado el código.

#### **8. Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información.**

Criterios de evaluación:

- a) Se han identificado las características de las bases de datos orientadas a
  - i) objetos.
- b) Se ha analizado su aplicación en el desarrollo de aplicaciones mediante
- j) lenguajes orientados a objetos.
- c) Se han instalado sistemas gestores de bases de datos orientados a
- k) objetos.
- d) Se han clasificado y analizado los distintos métodos soportados por los
- l) sistemas gestores para la gestión de la información almacenada.
- e) Se han creado bases de datos y las estructuras necesarias para el
- m) almacenamiento de objetos.
- n) Se han programado aplicaciones que almacenen objetos en las bases de datos creadas.
- f) Se han realizado programas para recuperar, actualizar y eliminar objetos de
- o) las bases de datos.
- g) Se han realizado programas para almacenar y gestionar tipos de datos
- p) estructurados, compuestos y relacionados.

#### **9. Gestiona información almacenada en bases de datos relacionales manteniendo la integridad y la consistencia de los datos.**

Criterios de evaluación:

- a) Se han identificado las características y métodos de acceso a sistemas
- q) gestores de bases de datos relacionales.
- b) Se han programado conexiones con bases de datos.
- c) Se ha escrito código para almacenar información en bases de datos.
- d) Se han creado programas para recuperar y mostrar información almacenada
- r) en bases de datos.
- e) Se han efectuado borrados y modificaciones sobre la información
- s) almacenada.
- f) Se han creado aplicaciones que ejecuten consultas sobre bases de datos.
- g) Se han creado aplicaciones para posibilitar la gestión de información presente
- t) en bases de datos relacionales.

#### **Metodología**

En primer lugar, se debe tener presente que el objetivo del ciclo formativo es preparar alumnos como futuros profesionales. Por tanto, se debe hacer hincapié en los contenidos procedimentales, si bien también es cierto que será necesario tener unos contenidos conceptuales de base. También es importante inculcar una serie de actitudes positivas y deseables en el trabajo, a través de los contenidos actitudinales. Las estrategias metodológicas a seguir por los profesores del módulo son:

- Método expositivo e interrogativo: consiste en el sistema clásico de enseñanza en que se imparten nuevos conocimientos, con el añadido de preguntar directamente a los alumnos para indagar en sus conocimientos previos, creando un vínculo entre los dos tipos de conocimientos.
- Método de aprendizaje por descubrimiento: consiste en proponer un problema a los alumnos, y que a través de unas indicaciones básicas sobre referencias bibliográficas y de consulta, sean capaces de encontrar la solución. Al haber intentado solucionar el problema, los alumnos que no lo consigan sí

entenderán mejor la solución del profesor, y estarán más motivados.

- Método de aprendizaje por proyectos: consiste en asignar proyectos de trabajo individuales o en grupo, en que los alumnos deben alcanzar unos objetivos.

Con todo ello, planteamos los siguientes grupos de actividades que se concretarán en cada unidad de trabajo.

- Actividades de introducción y motivación: para presentar un tema nuevo y captar el interés de los alumnos.
- Actividades de desarrollo: para profundizar en los contenidos de un tema.
- Actividades de recuperación: para ayudar a los alumnos con un ritmo más lento a alcanzar los mínimos exigibles.

Actividades de ampliación: para que los alumnos con un ritmo más rápido de la media de la clase, puedan profundizar en su aprendizaje manteniendo el interés en clase.

### Equipamientos y materiales.

#### Materiales

- Material documental y bibliográfico: bibliografía técnica específica, libros de texto, revistas de divulgación o profesionales, manuales de usuario en línea.
- Manual del alumno: recopilación de ejemplos, técnicas y procedimientos utilizados a lo largo de las sesiones, que constituirán una guía de referencia para su uso como recurso en el aula.
- Material audiovisual: proyector multimedia.
- Material didáctico de apoyo a la exposición del profesor: pizarra, presentaciones multimedia, apuntes elaborados por el docente, resúmenes.

#### - Equipamiento informático:

- Hardware: quince ordenadores, como mínimo, para el alumnado (al menos uno para cada dos alumnos) junto con el equipo o equipos del profesor, que ejercerá las funciones de servidor.

Impresora. Acceso a una red local.

- Sistema operativo: distribución de un sistema operativo Microsoft Windows XP/2008/7 y/o GNU/Linux debidamente configurado.

- Software: Entorno Integrado de Desarrollo: Eclipse, Netbeans, BlueJ, etc.

#### - Recursos TIC:

- Acceso a Internet. Supone un recurso básico en la enseñanza de esta materia, dado que se utiliza como fuente de información o para la obtención de aplicaciones, actualizaciones, herramientas online, etc. Además, tiene especial relevancia la inclusión del uso de Tecnologías de la Información en los módulos de Ciclos Formativos de la Familia Profesional de Informática y Comunicaciones, dado que están presentes en el entorno cotidiano del alumno y suponen un referente esencial tanto en su etapa de aprendizaje como en su futura actividad profesional.

## RELACIÓN DE UNIDADES DIDÁCTICAS

<b>1: Introducción a la Programación</b>		<b>Semana de inicio</b>	<b>1</b>
		<b>Nº horas teoría</b>	<b>5</b>
		<b>Nº horas práctica</b>	<b>5</b>
<b>RA</b>	<b>Contenidos</b>		
1	<p>Datos, algoritmos y programas. Paradigmas de programación. Lenguajes de programación. Herramientas y entornos para el desarrollo de programas. Errores y calidad de los programas.</p>		
	<b>Actividades Prácticas</b>		
	<p>Identificación y discusión de las distintas fases del ciclo de vida de una aplicación informática Diferenciación de los lenguajes interpretados de los lenguajes compilados Reconocimiento de los tipos básicos de datos que componen un programa Comparación de las distintas herramientas de diseño de algoritmos Aplicación de las distintas herramientas de diseño de algoritmos</p>		
	<b>Criterios de Evaluación</b>		
	<p>a) Se han identificado los bloques que componen la estructura de un programa informático. b) Se han creado proyectos de desarrollo de aplicaciones c) Se han utilizado entornos integrados de desarrollo. d) Se han identificado los distintos tipos de variables y la utilidad específica de cada uno. e) Se ha modificado el código de un programa para crear y utilizar variables. f) Se han creado y utilizado constantes y literales. g) Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje. h) Se ha comprobado el funcionamiento de las conversiones de tipo explícitas e implícitas. i) Se han introducido comentarios en el código.</p>		

<b>2: Identificación de los elementos de un programa informático</b>		<b>Semana de inicio</b>	<b>2</b>
		<b>Nº horas teoría</b>	<b>10</b>
		<b>Nº horas práctica</b>	<b>25</b>
<b>RA</b>	<b>Contenidos</b>		
3	<p>Estructura y bloques fundamentales.</p> <p>Identificadores.</p> <p>Palabras reservadas.</p> <p>Variables. Declaración, inicialización y utilización. Almacenamiento en memoria.</p> <p>Tipos de datos.</p> <p>Literales.</p> <p>Constantes.</p> <p>Operadores y expresiones. Precedencia de operadores</p> <p>Conversiones de tipo. Implícitas y explícitas (casting).</p> <p>Comentarios.</p> <p>Uso de Estructuras de Control:</p> <ul style="list-style-type: none"> <li>Estructuras de selección.</li> <li>Estructuras de repetición.</li> <li>Estructuras de salto.</li> </ul> <p>Control de excepciones.</p>		
	<b>Actividades Prácticas</b>		
	<p>Elaboración de algoritmos sencillos para practicar las estructuras de control, las variables elementales.</p> <p>Planteamiento de ejercicios</p> <p>Resolución de ejercicios en grupo</p> <p>Documentación de los ejercicios resueltos</p>		
	<b>Criterios de Evaluación</b>		
	<p>a) Se ha escrito y probado código que haga uso de estructuras de selección.</p> <p>b) Se han utilizado estructuras de repetición.</p> <p>c) Se han reconocido las posibilidades de las sentencias de salto.</p> <p>d) Se ha escrito código utilizando control de excepciones.</p> <p>e) Se han creado programas ejecutables utilizando diferentes estructuras de control.</p> <p>f) Se han probado y depurado los programas.</p> <p>g) Se ha comentado y documentado el código.</p>		



<b>3: Introducción a la orientación a objetos</b>		<b>Semana de inicio</b>	<b>6</b>
		<b>Nº horas teoría</b>	<b>10</b>
		<b>Nº horas práctica</b>	<b>25</b>
<b>RA</b>	<b>Contenidos</b>		
2,4	Principios básicos de la orientación a objetos. Concepto de clase. Estructura y miembros de una clase. Creación de atributos. Declaración e inicialización. Creación de métodos. Declaración, argumentos y valores de retorno. Creación de constructores. Características de los objetos. «Instanciación» de objetos, declaración y creación. Utilización de métodos. Parámetros y valores de retorno. Utilización de propiedades. Utilización de métodos estáticos. Constructores. Destrucción de objetos y liberación de memoria		
	<b>Actividades Prácticas</b>		
	Instalación del SDK por parte de cada alumno Instalación de un entorno de desarrollo gráfico para Java Utilización de ambas posibilidades en el desarrollo de aplicaciones Java Consulta de la documentación de los entornos de desarrollo vistos Diseño de clases sencillas		
	<b>Criterios de Evaluación</b>		
	a) Se han identificado los fundamentos de la programación orientada a objetos. b) Se han escrito programas simples. c) Se han instanciado objetos a partir de clases predefinidas. d) Se han utilizado métodos y propiedades de los objetos. e) Se han escrito llamadas a métodos estáticos. f) Se han utilizado parámetros en la llamada a métodos. g) Se han incorporado y utilizado librerías de objetos. h) Se han utilizado constructores. i) Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.		

<b>4: Utilización de objetos</b>		<b>Semana de inicio</b>	<b>10</b>
		<b>Nº horas teoría</b>	<b>5</b>
		<b>Nº horas práctica</b>	<b>15</b>
<b>RA</b>	<b>Contenidos</b>		
4	Utilización de clases y objetos. Almacenamiento en memoria. Tipos básicos vs. objetos. Destrucción de objetos y liberación de memoria.		
	<b>Actividades Prácticas</b>		
	Análisis de la abstracción y el encapsulamiento que proporcionan las clases diseñadas a la resolución del problema Creación de objetos a partir de las clases creadas		
	<b>Criterios de Evaluación</b>		
	a) Se ha reconocido la sintaxis, estructura y componentes típicos de una clase. b) Se han definido clases. c) Se han definido propiedades y métodos. d) Se han creado constructores. e) Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.		

<b>5: Desarrollo de clases</b>		<b>Semana de inicio</b>	<b>13</b>
		<b>Nº horas teoría</b>	<b>3</b>
		<b>Nº horas práctica</b>	<b>10</b>
<b>RA</b>	<b>Contenidos</b>		
4	Ámbito de atributos y variables. Sobrecarga de métodos. Visibilidad. Modificadores de clase, de atributos y de métodos. Paso de parámetros. Paso por valor y paso por referencia. Utilización de clases y objetos. Utilización de clases heredadas. Librerías y paquetes de clases. Utilización y creación. Documentación sobre librerías y paquetes de clases.		
	<b>Actividades Prácticas</b>		
	Convertir algoritmos sencillos elaborados previamente en pseudocódigo a programas fuentes de Java Utilización práctica del compilador y del depurador, a partir de los listados de programas fuentes anteriores Obtención y prueba del código interpretable por la máquina virtual.		
	<b>Criterios de Evaluación</b>		
	f) Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros. g) Se han definido y utilizado clases heredadas. h) Se han creado y utilizado métodos estáticos. i) Se han definido y utilizado interfaces. j) Se han creado y utilizado conjuntos y librerías de clases.		

<b>6: Estructuras de Almacenamiento</b>		<b>Semana de inicio</b>	<b>15</b>
		<b>Nº horas teoría</b>	<b>10</b>
		<b>Nº horas práctica</b>	<b>25</b>
<b>RA</b>	<b>Contenidos</b>		
6	Estructuras. Arrays unidimensionales y multidimensionales: Declaración. Creación de arrays unidimensionales y multidimensionales. Inicialización Acceso a elementos. Recorridos, búsquedas y ordenaciones. Cadenas de caracteres: Declaración. Creación de cadenas de caracteres. Inicialización Operaciones. Acceso a elementos, conversiones, concatenación.		
	<b>Actividades Prácticas</b>		
	Convertir algoritmos sencillos elaborados previamente en pseudocódigo a programas fuentes de Java, analizando las estructuras de datos necesarias. Diseño de la solución java de diferentes problemas, dividiendo las tareas entre diferentes clases usando las características de Java que se adecúen en cada caso. Documentación y comentario de las clases creadas.		
	<b>Criterios de Evaluación</b>		
	a) Se han escrito programas que utilicen arrays. b) Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados. f) Se han creado clases y métodos genéricos.		

<b>7: Herencia y Polimorfismo</b>		<b>Semana de inicio</b>	<b>18</b>
		<b>Nº horas teoría</b>	<b>5</b>
		<b>Nº horas práctica</b>	<b>20</b>
<b>RA</b>	<b>Contenidos</b>		
<b>7</b>	Composición de clases. Herencia. Superclases y subclases. Clases y métodos abstractos y finales. Sobreescritura de métodos. Constructores y herencia		
	<b>Actividades Prácticas</b>		
	Diseño de la solución java de diferentes problemas, dividiendo las tareas entre diferentes clases usando las características de Java que se adecúen en cada caso. Documentación y comentario de las clases creadas y su exportación al formato HTML mediante javadoc.		
	<b>Criterios de Evaluación</b>		
	b) Se han utilizado modificadores para bloquear y forzar la herencia de clases y métodos. c) Se ha reconocido la incidencia de los constructores en la herencia. d) Se han creado clases heredadas que sobrescriban la implementación de métodos de la superclase. e) Se han diseñado y aplicado jerarquías de clases. f) Se han probado y depurado las jerarquías de clases.		

<b>8: Estructuras avanzadas de datos: Colecciones de datos</b>		<b>Semana de inicio</b>	<b>22</b>
		<b>Nº horas teoría</b>	<b>5</b>
		<b>Nº horas práctica</b>	<b>25</b>
<b>RA</b>	<b>Contenidos</b>		
<b>6</b>	Tipos de colecciones (listas, pilas, colas, tablas...). Jerarquías de colecciones. Operaciones con colecciones. Acceso a elementos y recorridos. Uso de clases y métodos genéricos.		
	<b>Actividades Prácticas</b>		
	Elección de la colección más adecuada para cada caso Uso y aplicación de las diferentes colecciones de Java en los programas propuestos en clase.		
	<b>Criterios de Evaluación</b>		
	a) Se han identificado las estructuras avanzadas de datos. b) Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados. c) Se han utilizado listas para almacenar y procesar información. d) Se han utilizado iteradores para recorrer los elementos de las listas. e) Se han reconocido las características y ventajas de cada una de las colecciones de datos disponibles. f) Se han creado clases y métodos genéricos. g) Se han utilizado expresiones regulares en la búsqueda de patrones en cadenas de texto. h) Se han identificado las clases relacionadas con el tratamiento de documentos XML. i) Se han realizado programas que realicen manipulaciones sobre documentos XML.		

9: Lectura y Escritura de información		Semana de inicio	25
		Nº horas teoría	5
		Nº horas práctica	17
RA	Contenidos		
5	<p>Tipos de flujos. Flujos de bytes y de caracteres. Clases relativas a flujos. Utilización de flujos. Entrada desde teclado. Salida a pantalla. Ficheros de datos. Registros. Apertura y cierre de ficheros. Modos de acceso. Escritura y lectura de información en ficheros. Utilización de los sistemas de ficheros. Creación y eliminación de ficheros y directorios. Interfaces. Concepto de evento. Creación de controladores de eventos.</p>		
	<b>Actividades Prácticas</b>		
	<p>Creación de aplicaciones que necesiten un almacenamiento perdurable de datos Programación de los mecanismos adecuados para almacenar datos en ficheros.</p>		
	<b>Criterios de Evaluación</b>		
	<p>a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información. b) Se han aplicado formatos en la visualización de la información. c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas. d) Se han utilizado ficheros para almacenar y recuperar información. e) Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros. f) Se han utilizado las herramientas del entorno de desarrollo para crear interfaces gráficos de usuario simples. g) Se han programado controladores de eventos. h) Se han escrito programas que utilicen interfaces gráficos para la entrada y salida de información.</p>		

<b>10:Mantenimiento de la persistencia de los objetos</b>		<b>Semana de inicio</b>	<b>28</b>
		<b>Nº horas teoría</b>	<b>5</b>
		<b>Nº horas práctica</b>	<b>15</b>
<b>RA</b>	<b>Contenidos</b>		
8	Bases de datos orientadas a objetos. Características de las bases de datos orientadas a objetos. Instalación del gestor de bases de datos. Creación de bases de datos. Mecanismos de consulta. El lenguaje de consultas: sintaxis, expresiones y operadores. Recuperación, modificación y borrado de información. Tipos de datos objeto; atributos y métodos. Tipos de datos colección.		
	<b>Actividades Prácticas</b>		
	Modificación de los programas para almacenar los datos en bases de datos.		
	<b>Criterios de Evaluación</b>		
	a) Se han identificado las características de las bases de datos orientadas a objetos. b) Se ha analizado su aplicación en el desarrollo de aplicaciones mediante lenguajes orientados a objetos. c) Se han instalado sistemas gestores de bases de datos orientados a objetos. d) Se han clasificado y analizado los distintos métodos soportados por los sistemas gestores para la gestión de la información almacenada. e) Se han creado bases de datos y las estructuras necesarias para el almacenamiento de objetos. f) Se han programado aplicaciones que almacenen objetos en las bases de datos creadas. g) Se han realizado programas para recuperar, actualizar y eliminar objetos de las bases de datos. h) Se han realizado programas para almacenar y gestionar tipos de datos estructurados, compuestos y relacionados.		

<b>11: Gestión de bases de datos relacionales</b>		<b>Semana de inicio</b>	<b>31</b>
		<b>Nº horas teoría</b>	<b>10</b>
		<b>Nº horas práctica</b>	<b>15</b>
<b>RA</b>	<b>Contenidos</b>		
9	Establecimiento de conexiones. Recuperación de información. Manipulación de la información. Ejecución de consultas sobre la base de datos.		
	<b>Actividades Prácticas</b>		
	Programas para almacenar los datos en bases de datos, recuperar información y modificarla.		
	<b>Criterios de Evaluación</b>		
	a) Se han identificado las características y métodos de acceso a sistemas gestores de bases de datos relacionales. b) Se han programado conexiones con bases de datos. c) Se ha escrito código para almacenar información en bases de datos. d) Se han creado programas para recuperar y mostrar información almacenada en bases de datos. e) Se han efectuado borrados y modificaciones sobre la información almacenada. f) Se han creado aplicaciones que ejecuten consultas sobre bases de datos. g) Se han creado aplicaciones para posibilitar la gestión de información presente en bases de datos relacionales.		

## CRONOGRAMA

UD	Título	Semanas
1	Introducción a la programación	1
2	Identificación de los elementos de un programa informático	4
3	Introducción a la orientación a objetos	4
4	Utilización de objetos	3
5	Desarrollo de clases	2
6	Estructuras de almacenamiento	4
7	Herencia y polimorfismo	3
8	Estructuras avanzadas de datos: Colecciones de datos	4
9	Lectura y escritura de información	3
10	Mantenimiento de la persistencia de los objetos	3
11	Gestión de bases de datos relacionales	3

**Observaciones:**

Este cronograma podrá variar teniendo en cuenta el ritmo de aprendizaje de los alumnos, sus intereses, la duración de las pruebas, días festivos, etc.

## Procedimiento de Evaluación

### Procedimiento de Evaluación

Se utilizará un sistema de evaluación continua en el ámbito de un sistema docente y académico de tipo presencial, por lo que la asistencia a clase será obligatoria.

Dado que el enfoque de la metodología didáctica a emplear es fundamentalmente procedimental, la evaluación dará mucha importancia a la realización de las actividades que proponga el profesor.

### EVALUACIÓN SUMATIVA

El objetivo de esta evaluación es determinar el nivel de aprendizaje alcanzado por el alumno. En cada unidad de trabajo se han previsto actividades específicas de evaluación consistentes en:

- La respuesta a preguntas teóricas.
- La realización de programas, con la ayuda de recursos digitales y apuntes de clase, y con la posibilidad de realizar los mismos en el ordenador.

### EVALUACIÓN FORMATIVA

Pretende reconducir el proceso de aprendizaje para adaptarse al alumnado utilizando la observación como instrumento esencial, teniendo especial importancia en los siguientes momentos:

- **Control de asistencia de los alumnos**, teniendo en cuenta las faltas de cada alumno, que puede incidir de manera directa en el proceso continuo de enseñanza-aprendizaje.
- **Actividades de enseñanza-aprendizaje**: Cuando los alumnos realizan actividades de enseñanza-aprendizaje, el profesor va observando el trabajo que va desarrollando cada uno de los alumnos o grupo de ellos, lo que permite detectar aspectos no comprendidos o pendientes de superación. Ello permite dar directrices y explicaciones puntuales que ayuden a comprender mejor el problema y poder encontrar, consecuentemente, soluciones al problema planteado. Con este sistema se pretende, a priori, que el alumno pueda llegar de forma autónoma e independiente a la resolución de los problemas planteados.
- **Entrevistas con alumnos** mientras el resto de compañeros realiza actividades prácticas, el profesor detecta posibles problemas de comprensión de los conceptos o de la filosofía del trabajo encomendado.

### CRITERIOS DE CALIFICACIÓN

Se realizará una prueba individual y escrita por evaluación, consistente en la realización de uno o varios programas propuestos, donde se valorará:

1. El análisis del problema.
2. El diseño de las clases.
3. La claridad y eficiencia de los algoritmos.

La evaluación se considerará aprobada cuando la calificación sea igual o superior a 5.

A criterio del profesor se podrá pedir la realización de ejercicios para optar a la prueba individual de evaluación, en cuyo caso tendrán que ser entregados dentro del plazo establecido y con una calidad mínima.

Puesto que los contenidos del módulo son acumulativos, la nota final del módulo estará compuesta por el 20% de la calificación de la primera evaluación, más el 30% de la segunda, más el 50% de la tercera. Se considerará aprobado el módulo cuando aplicados los porcentajes a cada evaluación, su suma sea mayor o igual a 5.



#### **Procedimiento de Recuperación**

A lo largo del curso se establecerán actividades de recuperación y refuerzo de contenidos para los alumnos que no alcancen los objetivos mínimos de cada evaluación. Al ser un módulo en el que rige el criterio de evaluación continua, la recuperación de una evaluación se realizará compensando con la calificación obtenida en el resto de evaluaciones.

Los alumnos que por este procedimiento no alcancen la calificación de 5 o más, realizarán el examen ordinario de junio que contendrá los contenidos mínimos, de tal manera que así se podrá valorar si han adquirido o no las capacidades terminales para las que fue diseñado el módulo.

Con las mismas características también habrá otro examen de carácter extraordinario en junio para los alumnos que no hubieran aprobado de manera ordinaria.

#### **Actividades Extraescolares Previstas**

No existe ninguna actividad extraescolar prevista para este módulo. Las que se pudieran realizar serán las propuestas por el departamento en conjunto, a lo largo del curso y siguiendo las pautas establecidas por el Centro y la Familia Profesional, tratando de seleccionar aquellas que supongan un mayor aprovechamiento en relación al módulo que nos ocupa.

#### **Libro de Texto**

##### **LIBROS DE TEXTO:**

- **Fundamentos de Java 3ª edición**

- o Herbert Schildt

- o Mc Graw-Hill

- **Programación ( Ciclos formativos)**

- o Juan Carlos Moreno

- o Ra-Ma

##### **RECURSOS DIGITALES:**

<http://docs.oracle.com/javase/tutorial/>